METHOD AND COMPUTER PROGRAM FOR FIELD SPECTRUM OPTIMIZATION

invented by

Kenneth J. Ouimet

10580 E. Acacia Dr.

Scottsdale, AZ   85255

A citizen of the United States

Assigned To:

METHOD AND COMPUTER PROGRAM FOR FIELD SPECTRUM OPTIMIZATION

RELATED INVENTION

[0001]   The present invention claims priority under 35 U.S.C. §119(e) to: "*Enhanced Strategic Planning and Optimization System*," U.S. Provisional Patent Application Serial No. 60/459,925, filed 3 April 2003, which is incorporated by reference herein.

TECHNICAL FIELD OF THE INVENTION

[0002]   The present invention relates to the field of econometrics. More specifically, the present invention relates to enhanced processes and software for computing decisions for decision variables of a planning model characterizing an enterprise.

BACKGROUND OF THE INVENTION

[0003]   Enterprise modeling is the process of characterizing a real-world enterprise, using mathematical representations, graphs, and/or pictures. An enterprise planning model often utilizes mathematical modeling techniques to analyze complex real-world scenarios, typically with the goal of improving or optimizing performance. Accordingly, an enterprise planning model can ideally provide insight into past, current, and future operating performance, enabling managers to spot trends, identify opportunities, and affect outcomes.

[0004]   Multi-dimensional optimization entails determining a set of values that maximizes (or minimizes) a function of many decision variables. The types of mathematical relationships (for example, linear, nonlinear, or discontinuous) between the objective, the constraints, and the

decision variables determine how difficult the optimization is
to solve. The types of mathematical relationships also
determine the solution methods or algorithms that can be used
for optimization and the confidence that the solution is truly

5 optimal. Such multi-dimensional optimization is problematic in
that there is no known single multi-dimensional optimization
strategy that can tackle all problems in a satisfactory way.
In addition, the presence of constraints, even of simple ones,
enhances this difficulty.

10 [0005] One known strategy for performing multi-
dimensional optimization is an exhaustive search method in
which an entire configuration space of scenarios (i.e.,
possible combinations of the variables) is performed to select
an optimum out of all possibilities. The exhaustive search

15 method will yield the global optimum. Unfortunately, this
method is extremely computationally slow and therefore not
applicable in practical situations.

[0006] Other strategies entail the iterative gradient
search methods. These methods use information of the first and

20 possibly second order derivatives of the criterion function to
derive optimal search directions towards the optimum. Gradient
search methods guarantee decreasing criterion values in
successive iterations. The gradient search methods improve
significantly over the exhaustive search method, but are still

25 computationally slow and costly. A further disadvantage of
these methods is that they are sensitive to the initial
estimates of the unknowns if the criterion function has more
than one optimum. As such, the gradient search algorithm may
converge to a local optimum instead of the desired global

30 optimum. A global optimum is one in which are no other
feasible solutions with better objective function values. In
contrast, a local optimum is one in which there are no other

feasible solutions "in the vicinity" with better objective function values.

[0007] Much attention has been directed toward developing algorithms that circumvent convergence toward a local optimum. Two such algorithms are simulated annealing and genetic algorithms. The simulated annealing technique is essentially a local search, in which a move to an inferior solution is allowed with a probability that decreases as the process progresses. As such, there will always be a chance that a solution with a less good value might be retained in preference to a better solution. Thus, fine tuning of parameter settings is required.

[0008] Genetic algorithms are search techniques based on an abstract model of Darwinian evolution. Solutions are represented by fixed length strings over some alphabet ("gene" alphabet). Each string can be thought of as a "chromosome". The value of the solution represents the fitness of the chromosome. Survival of the fittest principle is then applied to create a new generation with slow increase of average fitness. Accordingly, genetic algorithms also have the facility of allowing some weak members to survive in the solution pool, but typically have mechanisms for favoring fitter solutions.

[0009] Both simulated annealing and genetic algorithms have a fair chance of circumventing convergence toward a local optimum. In addition, both methods are much faster than the exhaustive search method. However, both methods are still computationally expensive as compared with gradient search methods.

[0010] Another problem with conventional optimization algorithms arises when optimizing in the presence of coupled decision variables. As used herein, the term "coupled" refers

to decision variables and other objectives within an enterprise planning model that are connected causally to influence one another. This coupling further complicates optimization problems, leading to even more computational expense.

5   [0011]   Thus, what is needed is a technique, within an enterprise planning model, that can efficiently and cost effectively solve complex optimization problems.

SUMMARY OF THE INVENTION

10   [0012]   Accordingly, it is an advantage of the present invention that a method and computer program are provided for enhanced optimization calculation within a planning model.

[0013]   It is another advantage of the present invention that a method and computer program are provided for

15   computationally efficient optimization calculation of multi-dimensional planning models.

[0014]   Another advantage of the present invention is that a method and computer program are provided that mitigate the computational difficulties associated with the coupling

20   between decision variables within the planning model.

[0015]   The above and other advantages of the present invention are carried out in one form in a planning model characterizing an enterprise, by a method of computing decisions for a set of decision variables. The method calls

25   for generating a planning function representative of the planning model, the planning function depending upon the set of decision variables. The planning function is separated into independent planning functions, each of the independent planning functions depending upon different ones of the set of

30   decision variables. The method further calls for independently optimizing each of the independent planning functions to obtain the decisions for the different ones of the set of decision

variables, and presenting an outcome of the optimizing operation, the outcome indicating the obtained decisions.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] A more complete understanding of the present invention may be derived by referring to the detailed description and claims when considered in connection with the Figures, wherein like reference numbers refer to similar items throughout the Figures, and:

[0017] Fig. 1 shows a block diagram of an exemplary computing environment within which the method of the present invention may be practiced;

[0018] FIG. 2 shows a flow chart of a decision variable optimization process in accordance with a preferred embodiment of the present invention;

[0019] FIG. 3 shows a table of exemplary strategic constraint factor values specified in connection with the execution of the decision variable optimization process;

[0020] FIG. 4 shows a flow chart of an optimization subprocess in accordance with a preferred embodiment of the present invention;

[0021] FIG. 5 shows a graph of an optimum pricing envelope displaying profit (primary objective) versus dollar sales (strategic objective);

[0022] FIG. 6 shows an exemplary table of optimized decisions for decision variables of an exemplary scenario computed through the execution of the decision variable optimization process;

[0023] FIG. 7 shows a graph of an optimum pricing band displaying profit (primary objective) versus dollar sales (strategic objective) and volume (strategic objective);

[0024]   FIG. 8 shows a flow chart of an embedded constraint subprocess of the present invention;

[0025]   FIG. 9 shows a table of exemplary embedded constraint factor values specified in connection with the execution of the embedded constraint subprocess;

[0026]   FIG. 10 shows a table depicting a first exemplary planning model derived through the execution of the decision variable computation process of FIG. 2;

[0027]   FIG. 11 shows a table depicting a second exemplary planning model derived through the execution of the decision variable computation process;

[0028]   FIG. 12 shows a table depicting a third exemplary planning model derived through the execution of the decision variable computation process; and

[0029]   FIG. 13 shows a table depicting a fourth exemplary planning model derived through the execution of the decision variable computation process.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0030]   The following is a glossary of terms used herein:

[0031]   Enterprise: Any public, private, or governmental organization that provides items to be consumed by others, whether or not for profit. Presumably, an enterprise competes with other enterprises for the attention of customers and potential customers.

[0032]   Items: Are products in the form of goods, services, or a combination of goods and services. Items are broadly defined so that the same good and/or service provided in different market segments may be considered different items within the present context. Moreover, items are considered to be consumed within the present context when physically and/or

legally transferred to the customer, such as when a transaction occurs.

[0033]   Planning Model: A user-defined characterization of a real-world enterprise.

[0034]   Planning Function: A mathematical representation of a planning model. The planning function can include of set of related mathematical functions whose purpose is to simulate the response of the enterprise being modeled. The planning function can be linear, consisting of solely linear functions, or nonlinear, involving one or more nonlinear functions. The planning function can include primary objective functions and strategic objective functions.

[0035]   Primary Objectives: Represent the desires/goals of the decision maker/designer for the enterprise - such as to maximize profit, minimize cost, and so forth.

[0036]   Primary Objective Function: Is a mathematical equation that represents one of the primary objectives. The primary objective function depends upon a set of decision variables.

[0037]   Strategic Objectives: Represent factors that the decision maker/designer seeks to consider in conjunction with the primary objectives. The strategic objectives do not represent a physical restriction on the enterprise, thus need not be met rigorously. Strategic objectives include, for example, dollar sales, revenue, price image, service level, risk, product availability, product selection, market share, and so forth.

[0038]   Strategic Objective Function: Is a mathematical equation that represents one of the strategic objectives. The strategic objective function depends upon a subset of the decision variables.

[0039]   Decision Variables: Are variables under the control of the decision maker that could have an impact on the solution of the problem of interest. Decision variables include, for example, price, promotion type, promotion date, promotion duration, promotional discount, purchase date, product location, purchasing quantity, shelf space, product assortment, and so forth.

[0040]   Linear Functions: Contain terms each of which is composed of only a single, continuous variable raised to (and only to) the power of one.

[0041]   Nonlinear Functions: Are those in which more than a single variable may appear in a single term, and the variables may be raised to any power.

[0042]   Continuous Functions: Are those in which "small" changes in the input produce "small" changes in the output.

[0043]   Discontinuous Functions: Are those in which "small" changes in the input can produce abrupt changes in the output.

[0044]   Those skilled in the art will appreciate that managers of an enterprise desire to understand and achieve the objectives of the enterprise. However, managers typically do not wish merely to achieve a particular primary objective, but may additionally wish to accommodate broader considerations, i.e., the strategic objectives, in conjunction with the primary objectives. The preferred embodiments of the present invention provide a useful tool for computing decisions for a set of decision variables of a planning model. Moreover, the present invention enables the economically efficient computation of decisions, while taking into account the relationship between primary objectives and strategic objectives of an enterprise.

[0045]   The present invention is designated "field spectrum optimization." The term "field" refers to a Lagrange multiplier that is employed in the present invention as a strategic factor and/or a constraint factor. In other words,
5   the Lagrange multiplier is generally analogous to a "field" as that term may be employed in physics. The term "spectrum" refers to the set of all instances of a given field, i.e., all values for a particular strategic factor or embedded factor. Thus, the field spectrum optimization methodology described
10   herein solves, i.e., optimizes, over the "spectrum" (set of values) defined for a "field" (strategic and/or constraint factor). Moreover, the field spectrum optimization methodology described herein decouples decision variables (discussed below) to further facilitate economically efficient optimization of
15   decisions.

[0046]   In the following discussion relating to FIGs. 1-12, each Figure's reference numerals are keyed-in to its respective Figure number, i.e., FIG. 1 has reference numerals in the 100's, FIG. 2 has reference numerals in the 200's, and
20   so forth.

- FIG. 1 -

[0047]   FIG. 1 shows a block diagram of an exemplary
25   computing environment 100 within which the method of the present invention may be practiced. FIG. 1 depicts a processor section 102 in communication with an input/output section 104 and a memory 106. Nothing prevents processor section 102, input/output section 104 and/or memory 106 from including
30   numerous subsections that may or may not be located near each other. Thus, computing environment 100 may be provided by any

of a vast number of general or special purpose computers and/or computer networks.

[0048]   Memory 106 represents any manner of computer-readable media, including both primary memory (e.g., semiconductor devices with higher data transfer rates) and secondary memory (e.g., semiconductor, magnetic, and/or optical storage devices with lower data transfer rates). Input/output section 104 represents any manner of input elements (i.e., a keyboard, mouse, etc.) and output elements (i.e., monitors, printers, etc.). Data and computer programs may be transferred in to or out from memory 106 through input/output section 104.

[0049]   Memory 106 is depicted as having a code section 108 and a data section 110. Those skilled in the art will appreciate that any distinction between sections 108 and 110 may be due merely to different types of data and need not be due to physically different types of memory devices. Code section 108 stores any number of the types of computer programs typically found on computers and/or computer networks. In addition, code section 108 includes a planning model computer program 112 that may be partitioned into a planning function definition code segment 114 and a planning function separation code segment 116.

[0050]   Code section 108 also includes optimization algorithm computer programs 118. Optimization algorithm computer programs 118 of code section 108 may include specific optimization algorithms and/or general optimization algorithms. Specific optimization algorithms include, but are not limited to, closed form analytical solution, one-dimensional maximization of continuous decision variables, one-dimensional maximization of discrete decision variables, a general multi-dimensional optimization method, or other user-defined optimization algorithms, known to those skilled in the art.

General optimization algorithms include, but are not limited to, ant algorithm, genetic algorithm, tabu algorithm, simulated annealing, branch and bound, and other general optimization algorithms known to those skilled in the art.

[0051]   Prior to being transferred to memory 106, computer programs 112 or 118 may have resided on a computer-readable medium 120. Computer-readable medium 120 represents any location or storage device from which computer programs may be accessed, including remote servers, CD ROMs, and the like. Computer programs 112 and 118, and code segments 114 and 116 thereof, provide computer software that instructs processor section 102 how to manipulate and process a planning function 122 representative of a planning model of an enterprise, and how to store the resulting solution of planning function 122. Planning function 122 generally includes one or more primary objective functions 124 that depend upon a set of decision variables 125. Planning function 122 optionally includes one or more strategic objective functions 126, each of which is coupled with a strategic constraint factor 128, and each of which also depends upon decision variables 125. Planning function 122 is described in greater detail in connection with FIG. 2.

- FIG. 2 -

[0052]   FIG. 2 shows a flow chart of a decision variable optimization process 200 in accordance with a preferred embodiment of the present invention. FIG. 2 provides a conceptual vision of the present invention for comprehensiveness of description. However, those skilled in the art will readily recognize that task flow may vary greatly from that which is presented herein in response to actual code

instructions of a computer program of the present invention. In addition, it should be understood that tasks described herein may be performed manually by a decision-maker or may be carried out, at least in part, within computing environment 100

5   by computer programs 112 and 118.

    [0053]   Process 200 facilitates computationally efficient optimization calculation of decision variables 125. In addition, process 200 mitigates the computational difficulties associated with the coupling between decision variables 125

10   within the planning model. For purposes of the present invention, it is assumed that one or more primary objectives and, optionally, one or more strategic objectives may be incorporated into a planning model characterizing an enterprise.

15   **[0054]**   Decision variable optimization process 200 begins with a task 202. At task 202, primary objective function 124 is defined. More specifically, a primary objective of the enterprise is mathematically modeled through primary objective function 124 at task 202. Primary objective function 124,

20   namely $V\{x\}$, is a function of a set of decision variables 125, namely $\{x\}$. For simplicity of illustration, the present invention is described in connection with a single primary objective function 124. However, the present invention can be readily expanded to include additional primary objectives of

25   the planning model. Consequently, task 202 may further define additional primary objective functions for additional primary objectives of the planning model.

    [0055]   Following task 202, a query task 204 determines whether the planning model includes one or more strategic

30   objectives. When no strategic objective is defined in the planning model, process control proceeds to a task 206 (discussed below). However, when query task 204 determines

that the planning model includes a strategic objective, process 200 proceeds to a task 208.

[0056]   At task 208, strategic objective function 126 is defined. More specifically, a strategic objective of the enterprise is mathematically modeled through strategic objective function 124 at task 208. The strategic objectives represent significant business decisions that may be made by the decision-maker that can affect the primary objective. Strategic objective function 126, namely STG$\{x\}$, is also a function of decision variables 125, namely $\{x\}$.

[0057]   A task 210, performed in connection with task 208, couples strategic objective function 124 with strategic constraint factor 128, namely λ. The constant, represented by λ, is a Lagrange multiplier employed to test the effect that the strategic constraint, modeled by strategic constraint function 126, can have on the planning model. That is, different values for strategic constraint function 126 adjust an influence that the strategic objective will have on the planning model.

[0058]   In response to task 210, a task 212 specifies values for strategic constraint factor 128. These values can be user-specified or the values may be generated automatically by computing environment 100. The result of task 212 may be a table of values associated with particular scenario identifiers, and will be described in greater detail in connection with FIG. 3.

[0059]   Following task 212, a query task 214 determines whether the planning model includes another strategic objective. When there is another strategic objective, representing another significant business decision that may be made by the decision-maker that can affect the primary objective, process 200 loops back to task 208 to define a

second strategic objective function, couple the second
strategic objective function with a second strategic constraint
factor, and specify values for the second strategic constraint
factor. Subsequent tasks in the flowchart of FIG. 2 shall be
5   discussed herein below, following discussion of FIG. 3.

- FIG. 3 -

[0060]   FIG. 3 shows a table 300 of exemplary strategic
10   constraint factor values 128 specified in connection with the
execution of decision variable optimization process 200. In a
hypothetical situation, the planning model includes two
strategic objectives. Accordingly, through the execution of
process 200, two strategic objective functions 124 are defined.
15   Similarly, values for two strategic constraint factors 128
(i.e., two fields), namely λ_1 and λ_2, are specified. In table
300, each of a number of strategic constraint scenarios 302 are
given a unique strategic constraint scenario identifier 304.
Constraint factor values 306 for a first one of strategic
20   constraint factors 128, λ_1, are specified as the set (i.e.,
spectrum) including 0, 0.2, 0.4, and 0.6. Constraint factor
values 308 for a second one of strategic constraint factors
128, λ_2, are specified as the set (i.e., spectrum) including 0,
0.25, 0.5, 0.75, and 1. Strategic constraint scenarios 302
25   represent every possible combination of constraint factor
values 306 and 308. Thus, an exemplary strategic constraint
scenario 302', represented by strategic constraint scenario
identifier 304 of "G" reveals $\lambda_1=0.2$ and $\lambda_2=0.25$. Optimization
(discussed below) will be performed for each of strategic
30   constraint scenarios 302.

- FIG. 2 continued -

[0061]   Referring back to decision variable optimization process 200, when query task 214 determines that there are no further strategic objectives defined within the planning model, process control proceeds to task 206. Similarly, as described above, when query task 204 determines that the planning model does not include strategic objectives, process control proceeds to task 206.

[0062]   Task 206 combines primary objective function 124 and strategic objective function(s) 126, multiplied by their unique strategic constraint factors 128, to generate planning function 122. The generalized planning function 122 is represented as follows:

$$SP = V(\{x\}) + \sum_{s=1}^{S} \lambda_s STG_s(\{x\}) \tag{1}$$

where "S" is the number of strategic objectives. Equation (1) may also be subject to a set of tactical constraints (not shown). The tactical constraints operate as decision-level constraints with a possible strategic importance. Exemplary tactical constraints include a maximum or a minimum price for an item or class of items, and a defined relationship between prices, for example, the price of item 'n' must be less than or equal to the price of item 'm'. Overall ceilings or floors can also be set for tactical constraints. For example, the system can be constrained so that overall price change is less than a give percentage.

[0063]   Following task 206, process 200 proceeds to a query task 216. Query task 216 determines whether any of decision variables 125 are coupled in primary objective

function 124 of planning function 122. That is, query task 216 determines whether any decision variables 125 in primary objective function 124 are connected causally to influence one another. Examples of coupled decision variables 125 include the effects of competing products and stores; the effects of available demand to different purchasing decisions; spatial and temporal dependencies; the effect of price of a brand to a customer's choice of a brand; and so forth. The determination of coupling between decision variables 125 in primary objective function 124 of planning function further complicates the optimization calculation of decision variables 125.

[0064]   When query task 216 determines that decision variables 125 are coupled, process 200 proceeds to a task 218. Task 218 causes an embedded constraint subprocess to be performed. The embedded constraint subprocess is described in detail below in connection with FIGs. 8-9. The determination of coupling between decision variables 125 in primary objective function 124 of planning function further complicates the optimization calculation of decision variables 125. As will be discussed below, the embedded constraint subprocess mitigates the computational difficulties associated with this coupling.

[0065]   Following the execution of the embedded constraint subprocess initiated at task 218, program control proceeds to a task 220. Similarly, when query task 216 determines that decision variables 125 are not coupled, program control also proceeds to task 220. At task 220, an optimization algorithm is selected from the group of optimization algorithm computer programs 118. As discussed above, optimization algorithm computer programs 118 of code section 108 may include specific optimization algorithms and/or general optimization algorithms. The particular optimization

algorithm selected will depend upon the structure of planning function 122.

[0066]   A task 222 is performed in response to task 220. At task 222, planning function 122 is separated into independent planning functions in order to simplify the optimization calculation. In a preferred embodiment, each of the independent planning functions depends upon a different set of decision variables 125. Once separated, these independent planning functions can thus be treated as a sum of independent planning models, one for each item, as follows:

$$SP = \sum_i SP_i(p_i, \lambda) \qquad\qquad (2)$$

where the independent planning functions are represented by SP_i, and each independent planning function for item i (i.e., SP_i depends only upon a set of decision variables 125, represented by p_i. It should be understood that the set of decision variables 125 may include one or more of decision variables 125, although the total number of decision variables 125 is less than the total number of decision variables for the planning model represented by planning function 122.

Consequently, a generalized independent planning function can be presented as follows:

$$SP_i(x_i) = \max_{x_i} V_i(\{x_i\}) + \sum_{s=1}^{S} \lambda_s STG_s(\{x_i\}) \qquad\qquad (3)$$

where "S" is the number of strategic objectives, and the primary objective function, $V_i(x_i)$ is to be maximized for decision variable (x_i). The ability to separate planning function 122 into a number of smaller components, i.e., independent planning functions, for subsequent optimization

advantageously enhances the optimization process. Significant savings are realized in terms of required computing time and computing power by reducing a multi-dimensional planning function into lower order independent planning functions.

[0067]   Following task 222, a task 224 is initiated. Task 224 causes an optimization subprocess to be performed. The optimization subprocess is described in detail below in connection with FIG. 4. The optimization subprocess is performed to optimize a set of decision variables 125 for each of the independent planning functions, SP_i, of task 222. Subsequent tasks in the flowchart of FIG. 2 shall be discussed herein below, following discussion of FIG. 4.

- FIG. 4 -

[0068]   FIG. 4 shows a flow chart of an optimization subprocess 400 in accordance with a preferred embodiment of the present invention. Optimization entails solving the independent planning function, SP_i, to determine an optimum decision or set of decisions for that independent planning function, SP_i, at each scenario (for example, at each of strategic constraint scenarios 302). FIG. 4 provides a conceptual vision of the present invention for comprehensiveness of description. However, those skilled in the art will readily recognize that task flow may vary greatly from that which is presented herein in response to actual code instructions of a computer program of the present invention. In addition, tasks described herein may be performed manually by a decision-maker or may be carried out, at least in part, within computing environment 100.

[0069]   Optimization subprocess 400 begins with a task 402. A task 402 a "next" scenario is selected. For purposes

of the present invention strategic constraint scenarios 302 are
the scenarios selectable at task 402. It should also be noted
that at a first iteration of task 402, the "next" scenario 302
is a first one of the scenarios 302, in this case represented
by strategic constraint scenario identifier 304, "A".
Thereafter, the term "next" applies.

[0070]   A task 404 is performed in response to task 402.
At task 404, a "next" independent planning function, SP_i, is
selected. Again is should be noted that at a first iteration
of task 404, the "next" independent planning function is a
first one of the independent planning functions, SP_i,
determined at task 222 of decision variable optimization
process 200. Thereafter, the term "next" applies.

[0071]   Following task 404, a task 406 optimizes a set of
one or more decision variables 125 for the selected independent
planning function, SP_i, and the selected scenario 302.
Optimization calculations are performed utilizing the
optimization algorithm selected at task 220 of decision
variable optimization process 200.

[0072]   Following optimization task 406, a query task 408
is performed to determine whether there is another one
independent planning function, SP_i, for which optimization is
to be performed at the selected scenario. When there is
another independent planning function, SP_i, program control
loops back to task 404 to select the next independent planning
function and perform another optimization calculation for the
next independent planning function. However, when query task
408 determines that there are no further independent planning
functions, SP_i, for which optimization is to be performed,
optimization process 200 proceeds to a task 410.

[0073]   Task 410 causes the results of the optimization
calculations to be compiled for the selected scenario. As

mentioned above, optimization task 406 solves each independent
planning function, SP_i, to determine an optimum decision or
decisions for that independent planning function, SP_i, at the
selected scenario 302. Furthermore, as discussed above, the
5   independent planning functions, SP_i, can be treated as a sum of
independent planning models. Accordingly, an outcome of such
an iterative approach is a set of optimum decisions, $\{x^*\}$, for
each of decision variables 125 at the selected scenario 302.

[0074]   Once the optimum decisions $\{x^*\}$ are determined at
10   task 406, metrics that are functions of the optimum decisions
can be calculated. These metrics can include, for example,
financial metrics corresponding to the primary objective and
strategic metrics corresponding to the strategic objective(s).
The financial metrics can include, for example, gross profit,
15   net profit, and so forth. The strategic metrics can include,
for example, dollar sales, revenue, price image, service level,
risk, product availability, product selection, market share,
and so forth. Compilation of the results of the optimization
calculations may entail storing the results in database form in
20   data section 110 in associated with one of scenario identifiers
304 associated with the selected one of scenarios 302.

[0075]   Following task 410, a query task 412 is performed
to determine whether there is another one of scenarios 302 for
which optimization computations are to be performed. When
25   there is another one of scenarios 302, program control loops
back to task 402 to select the next one of scenarios 302, and
perform optimization calculations for the next one of scenarios
302. However, when query task 412 determines that there are no
further scenarios 302 at which optimization is to be performed,
30   optimization subprocess 400 exits.

- FIG. 2 continued -

[0076]   Referring back to decision variable optimization process 200, following the execution of optimization subprocess 400 at task 224, a task 228 is performed. At task 228, an outcome of optimization subprocess 400 is presented to the user via input/output section 104. As mentioned above, optimization subprocess 400 optimizes each independent planning function, SP_i, to determine an optimum decision or set of optimum decisions for that independent planning function, SP_i, at each scenario (for example, at each of strategic constraint scenarios 302). Furthermore, as discussed above, the optimization results, i.e., the optimum decisions, for all of the independent planning functions, are compiled at task 410, and metrics that are functions of the optimum decisions are calculated. Task 228 presents these metrics to a decision-maker. The metrics can be presented in a number of formats. For example, a textual list of optimized decisions and/or a variety of graphical representations may be displayed. Following task 228, decision variable optimization process exits.

- FIG. 5 -

[0077]   FIG. 5 shows an exemplary graph 500 of an optimum pricing envelope 501 displaying profit (primary objective) versus dollar sales (strategic objective). That is, graph 500 shows profit versus dollar sales, when decision variables 125 are prices, p_i, for items (i.e., goods, services, or a combination of goods and services offered by the enterprise). As discussed above, the independent planning functions can be treated as a sum of independent strategic models. Accordingly, graph 500 represents the sum of the independent planning

functions. Graph 500 shows that, starting from a very low
dollar sales, profits may increase as the aggregate of prices
(decisions for decision variables 125) increases. However, at
some point, further increase in the aggregate of prices causes
5 dollar sales to increase, but results in a drop in profit.

[0078] When the enterprise offers many different
products, a vast number of combinations of different pricing
scenarios may be devised. Each pricing scenario represents a
different mix of prices that may be offered for a set of
10 products being evaluated. Each scenario is represented by a
point 502 in graph 500, of which only a few of all possible
pricing scenario points 502 are shown. Only those pricing
scenario points 502 on envelope 501 are optimum pricing
scenarios. There are two regions of interest in graph 500.
15 Pricing scenario points 502 depicted in a region 504, which
reside within or underneath envelope 501, are inefficient.
Whereas, pricing scenario points 502 depicted in a region 506,
which reside outside envelope 501 are unachievable. By
selecting one of pricing scenario points 502 on envelope 501, a
20 user can be presented with the outcome, i.e., a set of
optimized decisions for decision variables 125.

Each of pricing scenario points 502 on envelope 501
represents a set of optimized decisions for decision variables
125 at one of strategic constraint scenarios 302. A decision-
25 maker may employ the outcome of the optimization calculations
as a guide to determine a preferred strategy, in this case, a
preferred pricing strategy, for the enterprise. For example, a
decision-maker may elect to sacrifice some profit to gain some
amount of dollar sales. If such is the case, the decision-
30 maker may select one of pricing scenario points 502 toward the
right side of envelope 501, such as an exemplary pricing
scenario point 502'. As readily illustrated in exemplary graph

500, some reduction in profit and a gain in dollar sales may occur. The decision-maker may determine that a short term reduction in profit, with increasing dollar sales, may be strategically important to the long term success of the
5   enterprise.

- FIG. 6 -

[0079]   FIG. 6 shows an exemplary table 600 of optimized
10   decisions 602 for decision variables 125 of exemplary pricing scenario point 502' computed through the execution of decision variable optimization process 200. Table 600 is illustrated for only one of pricing scenario points 502 for simplicity of illustration. However, it should be understood, that depending
15   upon which strategy (in this case, profit versus dollar sales) that a decision-maker elects, the decision-maker can select any of pricing scenario points 502 along envelope 501. Thus, a decision-maker may highlight any of pricing scenario points 502 along envelope 501 of graph 500, to obtain a set of optimized
20   decisions, such as, prices for items, promotions, and so forth, related to a particular strategic constraint scenario 302.

[0080]   Table 600 shows a list of items 604, uniquely identified by item identifiers 606, each being associated with one of decisions 602. In this case, the optimized decisions
25   602 are prices for each of items 604. Table 600 may also include non-price parameters 606 in association with items 604. Non-price parameters 608 may include timing, availability, promotion type, customer type, price thresholds, and other user-specified parameters of interest.

30

- FIG. 7 -

[0081]   FIG. 7 shows a graph 700 of an optimum pricing band 702 displaying profit versus dollar sales and volume. In this exemplary situation, two strategic constraints (dollar sales and value) may influence the outcome of the optimization. In such a situation, a general form for planning function 122 may be as follows:

$$SP = V(\{x\}) + \lambda_1 STG_1(\{x\}) + \lambda_2 STG_2(\{x\}) \qquad (4)$$

[0082]   The generalized planning function of equation (4) illustrates an interaction between primary objective function 124 and each of strategic objective functions 126. An optimization for each of strategic constraint scenarios 302 yields a series of optimum curves, each slightly shifted from one another. The result is optimum pricing band 702 of scenario points 704. Like graph 500, scenario points 704 depicted in a region 706, which reside within or underneath band 702, are inefficient. Whereas, scenario points 704 depicted in a region 708, which reside outside band 702 are unachievable. By selecting one of scenario points 704 within band 702, a decision-maker can be presented with the outcome, i.e., a set of optimized decisions for decision variables 125, when two strategic constraints have been considered, i.e., at one of strategic constraint scenarios 302.

- FIG. 8 -

[0083]   FIG. 8 shows a flow chart of an embedded constraint subprocess 800 of the present invention. FIG. 8 provides a conceptual vision of the present invention for

comprehensiveness of description. However, those skilled in the art will readily recognize that task flow may vary greatly from that which is presented herein in response to actual code instructions of a computer program of the present invention.

5   In addition, tasks described herein may be performed manually by a decision-maker or may be carried out, at least in part, within computing environment 100. It should be recalled that when query task 216 of optimization process 200 determines that decision variables 125 are coupled in primary objective

10   function 124, a task 218 initiates embedded constraint process 800.

[0084]   A generalized primary objective function 124 that includes a coupling between decision variables may be represented as follows:

15

$$V(\{p\}) = \sum_i \frac{h_i(p_i)g_i(p_i)}{\sum_k g_k(p_k)} \tag{5}$$

The function h_i could be any non-linear function of decision variables 125 for decision i. In this generalized primary

20   objective function 124, the decision variable 125 is the price of an item, and the denominator $\sum_k g_k(p_k)$ couples all of decision variables 125 together. Without this coupling, it would be possible to optimize each decision variable 125 independently. However, with this coupling, the optimization

25   becomes more complex. A standard technique for determining the set of prices {p} which would maximize (or minimize) primary objective function 124, V({p}), is to use some general optimization method. If the function is continuous, a gradient search may be employed. If the function is discontinuous,

30   simulated annealing, genetic algorithm, or some other general method may be employed. However, such algorithms are expensive

in terms of computational time and expense. Embedded constraint subprocess 800 is performed to reduce or eliminate this coupling of decision variables 125, thus mitigating computational difficulties and expense associated with the
5   coupling between decision variables.

[0085]   Embedded constraint subprocess 800 begins with a task 802. At task 802, an embedded constraint variable representing an embedded constraint is included in primary objective function 124. Utilizing the exemplary primary
10   objective function 124 of equation (5), the introduction of the embedded constraint variable is as follows:

$$V(\{p\}) = \sum_i \frac{h_i(p_i)g_i(p_i)}{Z} \tag{6}$$

15   The embedded constraint variable "Z" is introduced as a new variable to remove the coupling. However, so that the functional form of the original primary objective function 124 remains unchanged, the variable "Z" must satisfy an embedded constraint function.
20   [0086]   A task 804, performed in connection with task 802 defines an embedded constraint function. The embedded constraint function is defined to include the embedded constraint variable "Z" as follows:

25    $$\frac{\sum_{ik} g_k}{Z} - 1 = 0 \tag{7}$$

[0087]   Next, a task 806 couples the embedded constraint function with a Lagrange multiplier, i.e., an embedded constraint factor, γ, and a task 808 includes the embedded
30   constraint function coupled with the embedded constraint

factor, γ, with primary objective function 124 of planning function 122. Thus, primary objective function 124 becomes an effective objective function $\widetilde{V}$, as follows:

$$5 \qquad \widetilde{V}(\{p\}, Z) = \sum \frac{h_i(p_i)g_i(p_i)}{Z} + \gamma \left(\frac{\sum_k g_k(p_k)}{Z} - 1 \right) \qquad (8)$$

[0088]   As such, the objective function V({p},Z) can be maximized (or minimized) while satisfying its constraint by introducing embedded constraint factor, γ, and defining an

10   effective primary objective function $\widetilde{V}$. This transformation has allowed the coupling between decision variables 125 to be broken. Each optimum p^* can now be determined independently by maximizing the following:

$$15 \qquad p_i^*(\gamma): \max(h_i(p_i)g_i(p_i) + \gamma g_i(p_i)) \qquad (9)$$

[0089]   Following task 808, a task 810 specifies values for the embedded constraint factor, γ. Embedded constraint factor, γ, can be determined in a number of ways. If the

20   effective primary objective function is continuous, a gradient method such as Newton's Method could be employed. If the effective primary objective function is discontinuous, a bisection, grid search, or other discontinuous method may be employed. Alternatively, an estimate for the values of γ may

25   be determined by setting:

$$\left. \frac{\partial \widetilde{V}(\{p\}, Z)}{\partial Z} \right|_{p^*} = 0 \qquad (10)$$

where p^* represents a set of optimized decisions (for example, prices), which leads to

$$\gamma = \frac{-h_i(p_i{}^*)g_i(p_i{}^*)}{\sum_k g_k(p_k)} \tag{11}$$

5

Since the optimum decision (such as, optimum prices) are not known, estimates of the optimum decisions can also be used to provide a good estimate of the embedded constraint factor, γ. The result of task 810 may be a table of values associated with
10   particular scenario identifiers, and will be described in greater detail in connection with FIG. 9.

[0090]   Following task 810, embedded constraint subprocess 800 exits, and program control returns to task 222 of decision variable optimization process 200. Given the
15   strategic constraint factor γ, each optimum decision may be readily computed, and given the optimum decision, the original objective function, $V(\{p^*(\gamma)\})$, can be calculated. Consequently, a multi-dimensional optimization problem can be reduced to a 1-dimensional optimization problem $\tilde{V}(\gamma)$:

20

$$\tilde{V}(\gamma) = \frac{-h_i(p_i{}^*)g_i(p_i{}^*)}{\sum_k g_k(p_i{}^*)} \tag{12}$$

- FIG. 9 -

25   [0091]   FIG. 9 shows a table 900 of exemplary embedded constraint factor values 902 specified in connection with the execution of the embedded constraint subprocess 800. Through the execution of embedded constraint subprocess 800, an embedded constraint function, represented by equation (7) is

defined, and values for an embedded constraint factor 903
(i.e., a field), namely γ, is specified. In table 900, each of
a number of embedded constraint scenarios 904 are given a
unique embedded constraint scenario identifier 906. Embedded
constraint factor values 902 for embedded constraint factor 903
are specified as the set (i.e., spectrum) shown in table 900.
Thus, embedded constraint scenarios 904 represent each of
embedded constraint factor values 902.

[0092] Although not shown herein, it should be
understood that a table may be generated that includes both
strategic constraint factors 128 and embedded constraint factor
903. In such an instance, the resulting scenarios (not shown)
would represent every possible combination of constraint factor
values (such as those shown in FIG. 3), and embedded constraint
factor values (such as that shown in FIG. 9).

- FIGs. 10-13 -

[0093] FIGs. 10-13 provide tables of exemplary planning
models and illustrate the enhanced optimization capabilities of
the above described processes. The examples described in
connection with FIGs. 10-13 are for illustrative purposes only.
The present invention may be readily utilized to solve many
complex optimization problems by separating multi-dimensional
optimization problems into smaller components, as discussed
above.

[0094] The notation commonly employed in each of the
exemplary planning models of FIGs. 10-13 to represent
functions, variables, and other parameters, is listed below for
clarity of illustration:

$V(\{\_\})$ - primary objective function

STG($\{\_\}$) – strategic objective function

EMB($\{\_\}$) – embedded constraint function

SP($\{\_\}$) – planning function, also known as a strategic planning function when strategic objectives are included in the planning model

SP_i($\{\_\}$) – independent planning function

p_i – price of item i

c_i – cost of item i

p_i^* – optimized price of item i

p_i^c – current price of item i

λ – Lagrange multiplier utilized as a strategic constraint factor

γ – Lagrange multiplier utilized as an embedded constraint factor

Z – embedded constraint

h($\_$) – non-linear function

g($\_$) – function

US_i – unit sales of item i

q_i^o – demand parameter for a demand model

β_i – demand parameter for a demand model

AC_i – activity cost for item i

$\delta(x)$ – is a generalized function having the value 0 except at 0. In particular, $\delta(x)=1$ if $x=0$, and $\delta(x)=0$ if $x\neq 0$.

D – demand

ms_i – market share of item i

[0095]   Those skilled in the art will recognize that the nomenclature can alter greatly from that which is shown herein

depending upon the parameters of the particular optimization problem to be solved.

- FIG. 10 -

5

[0096]   FIG. 10 shows a table 1000 depicting a first exemplary planning model 1002 derived through the execution of decision variable computation process 200. First exemplary planning model 1002 includes a primary objective 1004, gross
10   profit, and a strategic objective 1006, dollar sales. Decision variables 1008 are defined as being prices of items, i. However, first exemplary planning model 1002 is not limited by tactical constraints.

[0097]   As shown, a primary objective function 1010 is
15   defined as being a function of the unit sales, US_i, of item, i, the price, p_i, of item, i, and the cost, c_i, of item, i. The unit sales has an exponential dependency on price, as shown, subject to demand parameters of a demand model. A strategic objective function 1012 is revenue or dollar sales, and is
20   defined as being a function of the price, p_i, of item, i, and the unit sales, US_i. Primary objective function 1010 and strategic objective function 1012, coupled with a strategic constraint factor 1014, are combined to yield a planning function 1016.

25   [0098]   A selected optimization algorithm 1018 is a closed form analytical algorithm for independently optimizing each of independent planning functions 1020. More specifically, independent planning functions 1020 yield an optimization equation 1022 that is readily calculated directly
30   to determine an optimum price, p_i^*, of item i.

- FIG. 11 -

[0099]   FIG. 11 shows a table 1100 depicting a second exemplary planning model 1102 derived through the execution of decision variable computation process 200. Second exemplary planning model 1102 represents a non-linear model with continuous decision variables. Accordingly, model 1102 is more complex than model 1002.

[0100]   Second exemplary planning model 1102 includes a primary objective 1104, gross profit, and a strategic objective 1106, dollar sales. Decision variables 1108 are defined as being prices of items, i. However, second exemplary planning model 1102 is not limited by tactical constraints. As shown, a primary objective function 1110 is defined as being a function of the unit sales, US_i, of item, i, the price, p_i, of item, i. Moreover, the price, p_i, is subject to a non-linear component, i.e., function h. In this example, second exemplary planning model 1102, derived through the execution of process 200, decouples the dependency of the prices.

[0101]   A strategic objective function 1112 is revenue or dollar sales, and is defined as being a function of the price, p_i, of item, i, and the unit sales, US_i. Primary objective function 1110 and strategic objective function 1112, coupled with a strategic constraint factor 1114, are combined to yield a planning function 1116.

[0102]   A selected optimization algorithm 1118 is a one-dimensional optimization algorithm, such as, an exhaustive search, Newton's method, and so forth, for continuous variables for independently optimizing each of independent planning functions 1120. Thus, independent planning functions 1120 yield an optimization equation 1122 that can be computed to determine an optimum price, p_i^*, of item i that maximizes the particular one of independent planning functions 1120.

- FIG. 12 -

[0103]   FIG. 12 shows a table 1200 depicting a third
exemplary planning model 1202 derived through the execution of
decision variable computation process 200.   Third exemplary
planning model 1202 represents a non-linear model with discrete
decision variables.   Accordingly model 1202 is more complex
than either of models 1002 and 1102.

[0104]   Third exemplary planning model 1202 includes a
primary objective 1204, net profit, and a strategic objective
1206, dollar sales.   Decision variables 1208 are defined as
being prices of items, i, and are discrete.   Third exemplary
planning model 1202 is also subject to tactical constraints
1209.   Tactical constraints 1209 require each price, p_i, to be
within minimum and maximum boundaries for each price.

[0105]   As shown, a primary objective function 1210
representing net profit is generally defined as being a
function of gross profit minus activity costs.   In particular,
primary objective function 1210 is defined as being a function
of the unit sales, US_i, of item, i, and the price, p_i, of item,
i.   However, the price, p_i, is subject to a non-linear
component, i.e., function h.   In addition, primary objective
function 1210 is affected by activity cost, AC_i associated with
changing the price of item i plus the cost of an employee
finding the item and changing its shelf price.   Through the use
of the delta function, $\delta(x)$, when there is no change in the
price of item i, there is no associated activity cost.
However, when the price of item i is adjusted, the activity
cost affects the gross profit to yield net profit.   A strategic
objective function 1212 is revenue or dollar sales, and is

defined as being a function of the price, p_i, of item, i, and the unit sales, US_i.

[0106]   Primary objective function 1210 and strategic objective function 1212, coupled with a strategic constraint factor 1214, are combined to yield a planning function 1216. Accordingly, planning function 1216 takes into account the cost to implement price changes to determine the net profit.

[0107]   A selected optimization algorithm 1218 is a one-dimensional optimization algorithm for discrete decision variables for independently optimizing each of independent planning functions 1220. The one-dimensional optimization is performed utilizing a simple search through all acceptable price points, p^a, and selecting the price, p_i^*, that maximizes independent planning functions 1220 as represented by an optimization equation 1222. As mentioned above, standard routines for solving one-dimensional optimization problems, include, but are not limited to Newton's method, Brent's method, golden section search in one-dimension, and exhaustive search over acceptable boundaries (for example, price points).

- FIG. 13 -

[0108]   FIG. 13 shows a table 1300 depicting a fourth exemplary planning model 1302 derived through the execution of decision variable computation process 200. Fourth exemplary planning model 1302 is provided to illustrate how process 200 employing embedded constraint subprocess 800 efficiently handle multinomial Logit type models. The present invention advantageously decouples the optimization in a non-linear planning model by introducing an embedded constraint, thereby decoupling the multinomial Logit form into a form, which can be solved analytically or using a one-dimensional optimization

method for discontinuous planning models. Although the decoupling of decision variables is illustrated in connection with multinomial Logit type models, embedded constraint subprocess 800 can be readily applied to other types of models.

5   [0109]   Fourth exemplary planning model 1302 includes a primary objective 1304, but does not include a strategic objective for simplicity of illustration. However, a strategic objective may be readily incorporated into fourth exemplary planning model 1302, as discussed above. Decision variables

10   1308 are defined as being prices of items, i. However, fourth exemplary planning model 1302 is not limited by tactical constraints.

[0110]   As shown, a primary objective function 1310 is defined as being a function of unit sales, US_i, of item, i;

15   price, p_i of item, i; and cost, c_i of item, i. However, unit sales is a function of demand, D, and market share, ms_i, of item, i. The market share is described by a multinomial Logit model that allocates the available demand to the different purchasing decisions. For instance, market share may describe

20   the customer choice between competing products or stores. The denominator, $\sum_k g_k$, within primary objective function 1310 couples decision variables 1308.

[0111]   In fourth exemplary planning model 1302, the function g_i reflects the exponential dependency of a utility

25   function (not shown) for item i. A utility function could take into account parameters that might influence the sales of item i in a store. These parameters include, for example, whether item i is on a display, whether it has signage, how much shelf space is allocated to the item, coupons, discounts, and so

30   forth.

[0112]   Through the execution of embedded constraint subprocess 800, an embedded constraint, Z, is introduced into

primary objective function 1310. The embedded constraint, Z, replaces the function $\sum_k g_k$ in primary objective function 1310. In addition, an embedded constraint function 1312 is utilized to enforce the variable Z. Primary objective function 1310 and

5   embedded constraint function 1312, coupled with an embedded constraint factor, γ, 1316 are combined to yield a planning function 1318. The inclusion of embedded constraint function 1312 with primary objective function 1310 serves to reduce the coupling between the decision variables of primary objective

10   function 1310. It should be apparent that since no strategic objective is defined in fourth exemplary planning model 1302, it also follows that no strategic objective function is provided in table 1300 for subsequent inclusion in planning function 1318.

15   [0113]   In fourth exemplary planning model 1302, a selected one-dimensional optimization algorithm 1320 may be a search algorithm for independently optimizing each of independent planning functions 1322. The one-dimensional optimization algorithm 1320 is performed to select the price,

20   p_i^*, that maximizes independent planning functions 1322 as represented by an optimization equation 1324 and satisfies the embedded constraint Z. That is, once values for each embedded constraint factor, γ, are specified at task 810 of embedded constraint subprocess 800, a suitable value for embedded

25   constraint factor, γ, is found that maximizes (or minimizes) primary objective function 1310 and concurrently satisfies the embedded constraint Z.

[0114]   Through the execution of embedded constraint subprocess 800, a coupled N-dimensional, nonlinear optimization

30   problem can be reduced to lower order independent planning functions. A lower order optimization problem yields a

significantly shorter processing time and greater stability of the optimization algorithm than prior art algorithms that handle coupled N-dimensional, nonlinear optimization problems.

[0115] In summary, the present invention teaches of method and computer program for enhanced optimization calculation within a planning model. The method and computer program produce computationally efficient optimization calculation by separating a multi-dimensional planning model into a set of independent planning models and determining optimal decisions for each of the independent planning models. Moreover, the method and computer program mitigate the computational difficulties associated with the coupling between decision variables within the planning model through the introduction of an embedded constraint.

[0116] Although the preferred embodiments of the invention have been illustrated and described in detail, it will be readily apparent to those skilled in the art that various modifications may be made therein without departing from the spirit of the invention or from the scope of the appended claims. For example, those skilled in the art will appreciate that the tasks depicted in FIGs. 2, 4, and 8 may be partitioned and sequenced in a wide variety of ways other than those specifically described here.